National Conference with International Participation
ENGINEERING MECHANICS 2003
Svratka, Czech Republic, May 12 – 15, 2003
paper no. 186

USING MODIFIED Q-LEARNING WITH LWR FOR INVERTED PENDULUM CONTROL

S. Věchet[1], **P. Miček**[2], **T. Březina**[3]

Summary: *Locally Weighted Learning (LWR) is a class of approximations, based on a local model. In this paper we demonstrate using LWR together with Q-learning for control tasks. Q-learning is the most effective and popular algorithm which belongs to the Reinforcement Learning algorithms group. This algorithm works with rewards and penalties. The most common representation of Q-function is the table. The table must be replaced by suitable approximator if use of continuous states is required. LWR is one of possible approximators. To get the first impression on application of LWR together with modified Q-learning for the control task a simple model of inverted pendulum was created and proposed method was applied on this model.*

1. Úvod

Dynamické úlohy regulace patří mezi vhodné kandidáty pro aplikaci opakovaně posilovaného učení(Reinforcement Learning - RL), i přesto, že většinou jde o úlohy se spojitými stavovými proměnnými. Mnoho algoritmů uvažuje diskrétní množiny stavů a akcí, které nejsou přímo aplikovatelné na tyto úlohy. Spojité proměnné jsou diskretizovány a tyto nové diskrétní hodnoty jsou použity pro RL. V těchto případech je nutné zvolit správnou diskretizaci stavů, resp. akcí aby bylo možno dosáhnout optimální řídící strategie(Březina & Krejsa & Věchet , 2002). Nejběžnějším způsobem reprezentace diskrétního stavového prostoru bývá tabulka Q-hodnot. V případě použití spojitého stavového prostoru tj. spojitých stavů a akcí je nutno tuto tabulku nahradit vhodným aproximátorem. Jako vhodný aproximátor byl zvolen algoritmus patřící do skupiny aproximátorů s lokálním modelem(Locally Weighted Learning - LWL)(Atkeson & Moore & Schaal, 1997). Z této skupiny tzv. paměťově orientovaných algoritmů byla pro jednoduchou implementaci zvolena metoda lokálně vážené regrese (Locally Weighted Regression - LWR).

2. Q-učení

Algoritmus Q-učení je založen na modelu agent-prostředí. Agent má k dispozici množinu akcí, kterými ovlivňuje stav prostředí. Vykonání akce a na systému nacházejícím se ve stavu s, má za

[1]Ing. Stanislav Věchet: ÚMT FSI VUT Brno; Technická 2; 616 69 Brno; e-mail: lio@email.cz
[2]Ing. Pavel Miček: ÚMT FSI VUT Brno; Technická 2; 616 69 Brno; e-mail: micek@umtn.fme.vutbr.cz
[3]RNDr. Ing. Tomáš Březina, CSc.: UAI FSI VUT Brno; Technická 2; 616 69 Brno; e-mail: brezina@uai.fme.vutbr.cz

následek, že systém přejde do nového stavu s' a dostane odměnu resp. trest r. Jeden ze způsobů implementace Q-funkce je tabulka Q-hodnot. V tomto případě je přepočtový vztah pro Q-funci daný

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma \max_a Q(s',a) - Q(s,a) \right] \tag{1}$$

Q-hodnota páru stav-akce (s,a) je aktualizována v závislosti na učícím poměru α, srážkovém faktoru γ a na maximální hodnotě Q-funkce, která je získána průchodem všech akcí pro nový stav s'.

3. Lokálně vážené učení

Koncept lokálně váženého učení (Locally Weighted Learning - LWL) je založen na aproximaci nelineárních funkcí počástech lineárním modelem, podobně jako Taylorův rozvoj prvního řádu. Ve velkém množství dostupných lokálních polynomů prokládaných daty, bývají nejčastěji používány lokální lineární modely. Hlavní problém LWL spočívá v nalezení oblasti kde můžeme lokálnímu modelu věřit. Oblast platnosti lineárního modelu je počítána z Gaussova jádra:

$$w_k = \exp(-\frac{1}{2}(\mathbf{x} - \mathbf{c}_k)^T \mathbf{D}_k (\mathbf{x} - \mathbf{c}_k)) \tag{2}$$

kde c_k je střed k-tého lineárního modelu, x bod vstupního vektoru v okolí c_k, D_k odpovídá pozitivně semidefinitní matici vzdálenosti určující velikost a tvar platné oblasti lineárního modelu, w_k reprezentuje váhu závislou na vzdálenosti bodů x a c_k, kdy vzdálenější bod má menší vliv na tvar proložené funkce než bod ležící blíže. Je možné použít jiné jádro funkce, což ovšem mění málo kvalitu.

4. Lokálně vážená regrese

Mezi nejjednodušší algoritmy s lokálním lineárním modelem patří lokálně vážená regrese (Locally weighted learning - LWR). Vstupní hodnoty pro odhad jsou množina p tréninkových bodů $\{x_i, y_i\}$, které mají být aproximovány, kde x_i jsou vstupy a y_i odpovídající výstupy a dále dotazovaný bod x_q pro který se má vypočítat odhad y_q, což je výstup tohoto algoritmu. Predikce pro dotazovaný bod x_q je generována následujícím algoritmem vážené regrese:

Algoritmus LWR
Dáno:
Dotazovaný bod x_q a p tréninkových bodů $\{x_i, y_i\}$
Výpočet predikce:
a) výpočet diagonální matici vah W
$w_{ii} = \exp(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_q)^T \mathbf{D} (\mathbf{x}_i - \mathbf{x}_q))$
b)sestavení matice X a vektoru y
$\mathbf{X} = (\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ...\tilde{\mathbf{x}}_p)^T$ kde $\tilde{\mathbf{x}}_i = [(\mathbf{x}_i - \mathbf{x}_q)^T 1]^T$
$\mathbf{y} = (y_1, y_2, ...y_p)^T$
c)výpočet lokálního lineárního modelu
$\beta = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{y}$
d)predikce pro x_q je:
$\hat{y}_q = \beta_{n+1}$ (3)

β_{n+1} značí (n+1)-tý element vektoru β. Výpočtová náročnost LWR je úměrná pn^2. Běžně je většina bodů z množiny p tréninkových dat aproximována nulovou vahou, protože jsou od dotazovaného bodu příliš vzdáleny což ve značné míře přispívá k redukci výpočtové náročnosti. LWR může být aplikováno na problémy v reálném čase pokud mají malý počet vstupů n.

Jediný neznámý parametr v algoritmu (3) je metrika vzdálenosti $\mathbf{D}$. Při značném množství dat, může být $\mathbf{D}$ optimalizováno vynecháváním dotazovaného bodu tzv. leave-one-out cross validation. Abychom se vyhnuli mnoha neznámým parametrům, předpokládáme $\mathbf{D}$ jako globální diagonální matici $\mathbf{D} = h \cdot diag([n_1, n_2, ..., n_n])$ kde h je měřítko a n_i normalizuje rozsah vstupní dimenze např. pro každý vstup rozptyl $n_i = 1/\sigma_i^2$. Optimalizace parametru $\mathbf{D}$ je realizována jednoduše jako jednorozměrné prohledávání přes parametr h:

Algoritmus optimalizace
Dáno:
množina H daná hodnotamy h_r
Algoritmus:
pro všechna $h_r \in H$:
$sse_r = 0$
pro všechna i=1:p
a)$\{\mathbf{x}_\mathrm{i}, y_i\}$
b) dočasně vyjmout $\{\mathbf{x}_\mathrm{i}, y_i\}$ z množiny tréninkových dat
c) výpočet LWR predikce $\hat{y}_q$ s redukovanými daty
d) $sse_r = sse_r + (y_i - \hat{y}_q)^2$
Zvolit optimální h^* jako $h^* = \min_r \{sse_r\}$ (4)

Samozřejmě je možné, ovšem za cenu zvýšené výpočtové náročnosti, posuzovat všechny parametry v matici vzdálenosti jako neznámé.

5. Trénovací algoritmus

Metodu LWR lze použít v Q-učení několika způsoby. Jednou z možností je použít LWR společně s tabulkou Q-hodnot, kdy slouží k odhadu Q-hodnoty pro vstupní spojité stavy a akce z diskrétní tabulky. Další možností je použít LWR aproximátoru přímo jako implementaci Q-funkce (Forbes & Andre , 2000). V tomto článku je popsán první z výše popsaných algoritmů kombinující LWR s tabulkou Q-hodnot. Pseudokód tohoto algoritmu je popsán algoritmem 5. Vstupní parametry jsou spojitá akce a stav (s, a), učící poměr α, srážkový faktor γ a parametr h určující maximální vzdálenost okolních bodů od dotazovaného bodu, které se ještě použijí pro odhad pomocí LWR. Nejprve vypočítáme odhad q pomocí LWR (krok 1), body použité pro aproximaci uložíme do matice K (krok 3). V dalším kroku spočítáme matici vah W v závislosti na vzdálenosti od dotazovaného bodu. Kroky 5 a 6 provádí aktualizaci původní hodnoty dotazovaného bodu v tabulce. Poslední krok aktualizuje okolní body v závislosti na

vahách vzdálenosti a na velikosti Δq.

Modifikovaný algoritmus Q-učení
Dáno:
pár stav-akce (s, a)
učící poměr α
srážkový faktor γ
LWR okolí h
Algoritmus:
1) $q \leftarrow Q_{odhad}(s, a)$ metoda LWR
2) $q' \leftarrow \max_a Q(s', a)$
3) $K \leftarrow$ použité body pro aproximaci q
4) $W_i \leftarrow \exp(-(\vec{q} - \vec{k_i})^2/h^2)$
5) $\Delta q \leftarrow \alpha\,(r + \gamma q' - q)$
6) $Q(s, a) \leftarrow Q(s, a) + \Delta q$
7) aktualizace pro každý bod $Q(s_i, a_i)$ v K
$Q(s_i, a_i) \leftarrow Q(s_i, a_i) + \Delta q w_i$ (5)

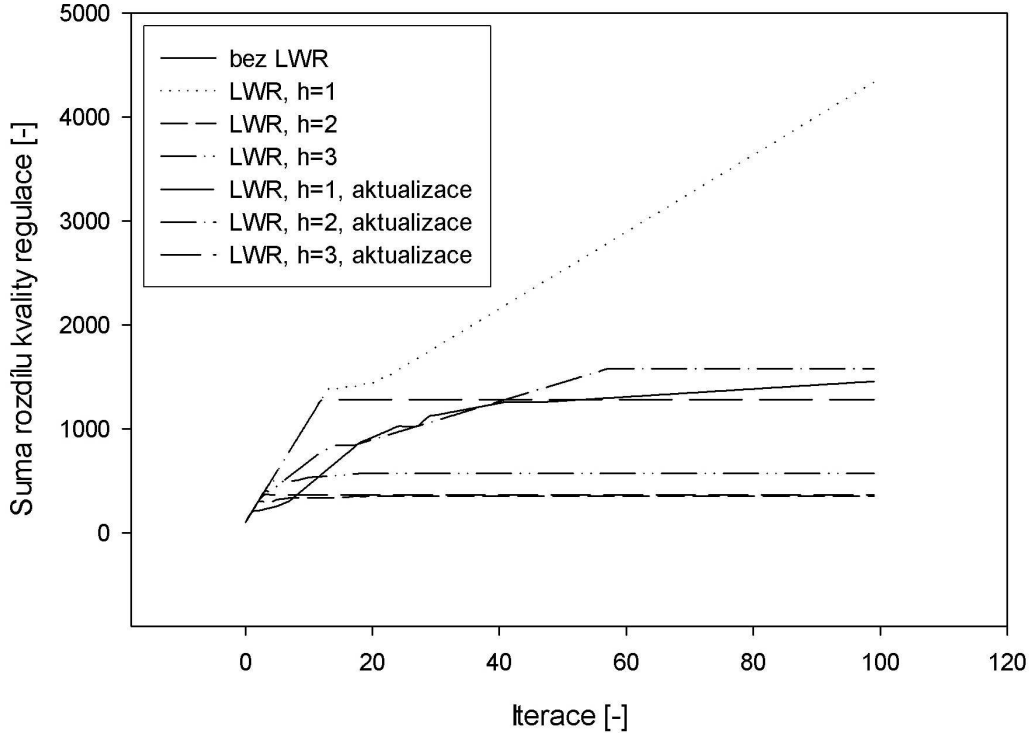
6. Provedené experimenty

Pro počáteční experimenty s LWR byl použit nenáročný model inverzního kyvadla. Byla zanedbána hmotnost vozíku a tření. Jako akce bylo použito zrychlení se kterým se pohybuje vozík a jako stavu úhel odklonu kyvadla od vertikální osy. Cílem agenta-řídícího členu je udržet kyvadlo ve vymezeném intervalu $< -\varphi_{\max}, \varphi_{\max} >$. Akce byly voleny z intervalu $< -a, a >$. Posilovací funkce stanovuje odměnu/trest z množiny $\{-1, 0, 1\}$ následujícím pravidlem

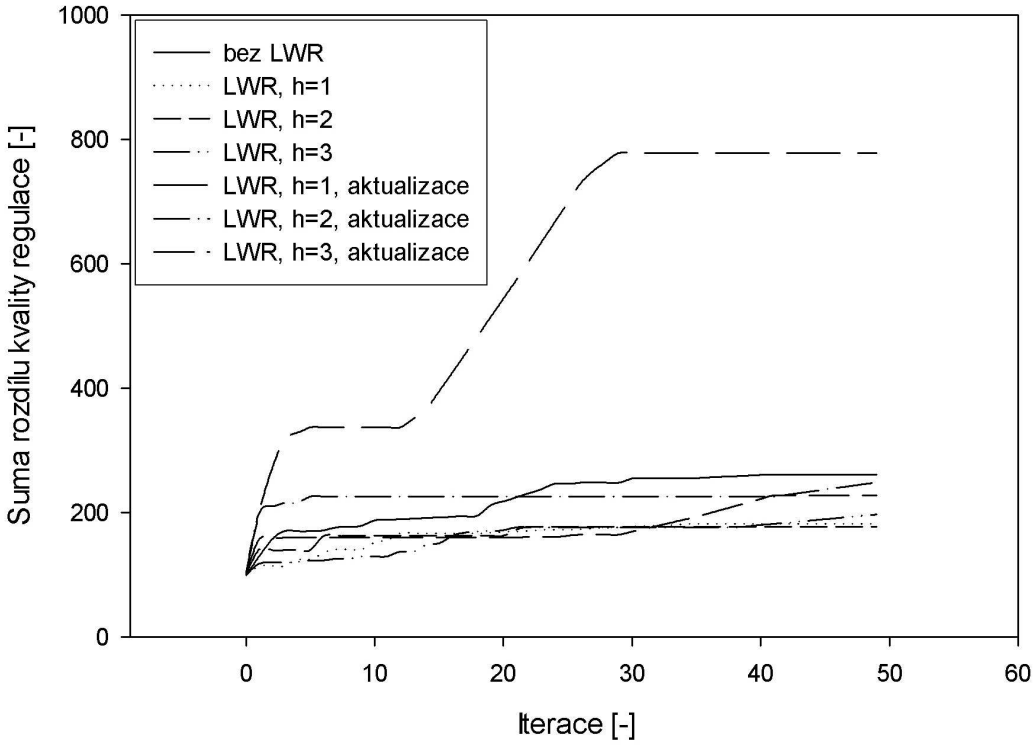
$$r = \left\{ \begin{array}{ll} 1 & \varphi \leq |\varphi_{\min}| \\ 0 & \varphi \in (|\varphi_{\min}|, |\varphi_{\max}|\rangle \\ -1 & jinak \end{array} \right\}$$

Pro experimenty bylo použito těchto parametrů modelu $m = 0.5kg$, $g = 9.81kgms^2$, $l = 0.2m$ a parametrů učení $a = 10ms^2$, $\varphi_{\min} = 0.1°$, $\varphi_{\max} = 20°$, $\alpha = 0.2$, $\gamma = 0.99$
V experimentech byl sledován vliv:
velikosti stavového prostoru
velikosti okolí h dotazovaného bodu pro odhad pomocí LWR
aktualizace bodů kroku 7, algoritmu 5.

Systém byl testován následujícím způsobem: Nejprve bylo vygenerováno náhodně n počátečních stavů. Tyto stavy byly použity během testování a bylo sledováno, zda po stanovený počet kroků dokáže agent udržet výchylku kyvadla v určeném intervalu. Pokud to agent dokázal, byl pokus vyhodnocen jako úspěšný. Tím byla stanovena procentuální úspěšnost v průběhu učení. Dále byla sledována hodnota sumy rozdílů stoprocentní úspěšnosti a skutečné úspěšnosti a to během celé doby učení viz. obrázek 1, 2.



Obrázek 1: Průběh učení pro tabulku 6x6



Obrázek 2: Průběh učení pro tabulku 20x20

7. Závěr

Abychom mohli použít spojitých stavů a akcí v konvenčním Q-učení je nutné použít vhodný aproximátor. Jako první krok při použití spojitých stavů a akcí jsme zvolili LWR. Algoritmus pro aproximaci Q-funkce patřící do skupiny LWL algoritmů. Zvolená metoda byla testována na jednoduchém simulační modelu inverzního kyvadla.

V prvních pokusech bylo LWR použito ve fázi učení pouze pro odhad Q-hodnoty při použití spojitých stavů. Q-funkce zůstala implementována jako tabulka a vypočtená hodnota byla použita pro změnu Q-hodnoty tabulky. Kvalita regulace takto upraveného regulačního členu je srovnatelná s běžným, pouze tabulkou imlementovaným, regulačním členem.

Další stupněm použití spojitých stavů a akcí je použit Q-učení, kde je tabulka Q-hodnot plně nahrazena metodou LWR. Metoda LWR je velmi citlivá na nastavení vlastních parametrů a ve spojitosti s parametry Q-učení, je velmi obtížné nastavit tyto jako optimální. Kvalita regulace takto modifikovaného regulačního členu je horší, než původního, který používá pouze diskrétní tabulku.

Výhody použití spojitých stavů a akcí jsou tak velké, že tato metoda bude podrobována dalšímu zkoumání. V dalších experimentech budou také prověřeny další zdokonalené algoritmy založené na LWL.

8. Poděkování

Práce vznikla za podpory pilotního projektu ÚT AV ČR č. 52020 „Řízení kráčivého robotu s využitím metod umělé inteligence“, výzkumného záměru MŠMT MSM 262100024 „Výzkum a vývoj mechatronických soustav“ a výzkumného záměru CEZ:J22/98:261100009 „Netradiční metody studia komplexních a neurčitých systémů“.

9. Literatura

Březina T. & Krejsa J. & Věchet S. (2002) *Stochastic Policy in Q-Learning Used for Control of AMB* Engineering Mechanics 2002, pp. 7-8.

Atkeson,C. G. & Moore,A. W. & Schaal, S. (1997) *Locally Weighted Learning* Artificial Intelligence Review, pp. 11-73.

Forbes J.& Andre D. (2000) *Practical Reinforcement Learning in Continuous Domains*, Computer Science Division, Tech. Rep. UCB/CSD-00-1109